

Week 1

# Welcome to the Course

---

© IBM Corporation. All rights reserved.

# Prerequisites

---

- Professionals with knowledge of Python programming language
- Aspiring data analysts
- Aspiring data scientists



# What you will learn

---



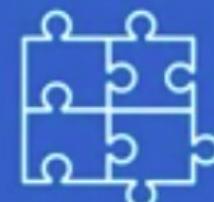
Create  
impactful data  
visuals



Explore  
advanced  
visualization  
tools



Identify  
different web-  
based  
dashboarding  
tools



Practice data  
visualization  
skills

# Module 1: Introduction to Data Visualization

---

- Best practices
- History and architecture of Matplotlib
- Basic plotting with Matplotlib
- Dataset on Canadian immigration
- Data analysis



## Module 2: Basic and Specialized Visualization Tools

---

- Diverse range of plots
  - Area plots,
  - Histogram
  - Bar charts
  - Pie charts
  - Box plots
  - Scatter plots
- Process of creating visualization tools using Matplotlib



## Module 3: Advanced Visualizations and Geospatial Data

---

- Waffle charts and word cloud
- Seaborn
- Folium
- Process of creating maps using Folium
- Creating Choropleth map using Folium



## Module 4: Creating Dashboards with Plotly and Dash

---

- Benefits of dashboards
- Different web-based dashboarding tools
- Plotly
  - Plotly graph objects
  - Plotly express
- Dash



## Module 5: Final Project and Exam

---

- Final lab assignment
- Analyze the automobile sales data



# Key takeaways

- Use different visualization libraries in Python
- Use basic and advanced visualization tools
- Identify different web-based dashboarding tools



# Making the most out of the course

---

- View every video
- Review your knowledge with each quiz
- Complete all the hands-on labs



## Welcome to the Course

- Video: Welcome to the Course

3 min

- Reading: How to Make the Most of this Course

5 min

- Reading: Syllabus

10 min

## Introduction to Data Visualization

# Syllabus

## Abstract

One of the most important skills of successful data scientists and data analysts is the ability to tell a compelling story by visualizing data and findings in an approachable and stimulating way. In this course you will learn many ways to effectively visualize both small and large-scale data. You will be able to take data that at first glance has little meaning and present that data in a form that conveys insights.

This course will teach you to work with many Data Visualization tools and techniques. You will learn to create various types of basic and advanced graphs and charts like: Waffle Charts, Area Plots, Histograms, Bar Charts, Pie Charts, Scatter Plots, Word Clouds, Choropleth Maps, and many more! You will also create interactive dashboards that allow even those without any Data Science experience to better understand data and make more effective and informed decisions.

You will learn hands-on by completing numerous labs and a final project to practice and apply the many aspects and techniques of Data Visualization using Jupyter Notebooks and a Cloud-based IDE. You will use several data visualization libraries in Python, including Matplotlib, Seaborn, Folium, Plotly & Dash.

## Course Learning Objectives





## Welcome to the Course

**Video:** Welcome to the Course  
3 min

**Reading:** How to Make the Most of this Course  
5 min

**Reading:** Syllabus  
10 min

## Introduction to Data Visualization

*After completing this course, a learner will be able to:*

- Implement data visualization techniques and plots using Python libraries, such as Matplotlib, Seaborn, and Folium to tell a stimulating story.
- Create different types of charts and plots such as line, area, histograms, bar, pie, box, scatter, and bubble.
- Create advanced visualizations such as waffle charts, word clouds, regression plots, maps with markers, & choropleth maps.
- Generate interactive dashboards containing scatter, line, bar, bubble, pie, and sunburst charts using the Dash framework and Plotly library.

## Module 1

### Title: Introduction to Data Visualization Tools

#### Description

Data visualization is a way of presenting complex data in a form that is graphical and easy to understand. When analyzing large volumes of data and making data-driven decisions, data visualization is crucial. In this module, you will learn about data visualization and some key best practices to follow when creating plots and visuals. You will discover the history and the architecture of Matplotlib. Furthermore, you will learn about basic plotting with Matplotlib and explore the dataset on Canadian immigration, which you will use during the course. Lastly, you will analyze data in a data frame and generate line plots using Matplotlib.





## Welcome to the Course

- Video:** Welcome to the Course  
3 min

- Reading:** How to Make the Most of this Course  
5 min

- Reading:** Syllabus  
10 min

## Introduction to Data Visualization

### Objectives

*By the end of this week, you will be able to:*

- Discuss data visualization and its importance
- Discover the history of Matplotlib and its architecture
- Use Matplotlib to create plots employing Jupyter notebook
- Explore the dataset on immigration to Canada
- Identify the steps to analyze data in Pandas data frame
- Use Matplotlib to create line plots

### Activities

#### Lesson 0: Welcome to the Course

- Welcome to the Course
- How to Make the Most of this Course
- Syllabus





## Welcome to the Course

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

## Introduction to Data Visualization

### Lesson 1: Introduction to Data Visualization

- Overview of Data Visualization
- Types of Plots
- Plot Libraries
- Introduction to Matplotlib
- Basic Plotting with Matplotlib
- Dataset on Immigration to Canada
- Line Plots
- Hands-on Lab: Exploring and Pre-processing a Dataset using Pandas
- Hands-on Lab: Introduction to Matplotlib and Line Plots
- Practice Quiz: Introduction to Data Visualization
- Module 1 Summary: Introduction to Data Visualization Tools
- Module 1 Cheat Sheet
- Module 1 Graded Quiz: Introduction to Data Visualization Tools





## Welcome to the Course

**Video:** Welcome to the Course  
3 min

**Reading:** How to Make the Most of this Course  
5 min

**Reading:** Syllabus  
10 min

## Introduction to Data Visualization

## Module 2

### Title: Basic and Specialized Visualization Tools

#### Description

Visualization tools play a crucial role in data analysis and communication. These are essential for extracting insights and presenting information in a concise manner to both technical and non-technical audiences. In this module, you will create a diverse range of plots using Matplotlib, the data visualization library. Throughout this module, you will learn about area plots, histograms, bar charts, pie charts, box plots, and scatter plots. You will also explore the process of creating these visualization tools using Matplotlib.

#### Objectives

*By the end of this week, you will be able to:*

- Explore an area plot with an illustration and create it using Matplotlib
- Define a histogram with an illustration and create it using Matplotlib
- Describe a bar chart with an illustration and create it using Matplotlib
- Discover a pie chart with an illustration and create it using Matplotlib





## Welcome to the Course

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

## Introduction to Data Visualization

- Describe a box plot with an illustration and create it using Matplotlib
- Discover a scatter plot with an illustration and create it using Matplotlib

## Activities

### Lesson 1: Basic Visualization Tools

- Area Plots
- Histograms
- Bar Charts
- Hands-on Lab: Area Plots, Histograms, and Bar Charts
- Practice Quiz: Basic Visualization Tools

### Lesson 2: Specialized Visualization Tools

- Pie Charts
- Box Plots
- Scatter Plots



**Welcome to the Course** **Video:** Welcome to the Course

3 min

**Reading:** How to Make the Most of this Course

5 min

**Reading:** Syllabus

10 min

**Introduction to Data Visualization**

- Hands-on Lab: Pie Charts, Box Plots, Scatter Plots, and Bubble Plots
- Plotting Directly with Matplotlib
- Hands-on Lab: Plotting Directly with Matplotlib
- Practice Quiz: Specialized Visualization Tools
- Module 2 Summary: Basic and Specialized Visualization Tools
- Module 2 Cheat Sheet
- Module 2 Graded Quiz: Basic and Specialized Visualization Tools

## Module 3

### Title: Advanced Visualizations and Geospatial Data

#### Description

Advanced visualization tools are sophisticated platforms that provide a wide range of advanced features and capabilities. These tools provide an extensive set of options that help create visually appealing and interactive visualizations. In this module, you will learn about waffle charts and word cloud including their application. You will explore Seaborn, a new visualization library in Python, and learn how to create regression plots using it. In addition, you will learn about folium, a data visualization library that visualizes geospatial data. Furthermore, you will explore the process of creating maps using Folium and superimposing them with markers to make them interactive. Finally,



**Welcome to the Course**

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

**Introduction to Data Visualization**

the process of creating maps using Folium and superimposing them with markers to make them interesting. Finally, you will learn how to create a Choropleth map using Folium.

**Objectives**

*By the end of this week, you will be able to:*

- Explore waffle charts and word cloud along with their application
- Describe Seaborn and explore the process of generating attractive regression plots
- Describe Folium and explore the process of creating maps
- Explore the process of superimposing markers on maps using Folium
- Describe Choropleth maps with the help of an illustration
- Explore the process of creating a Choropleth map using Folium

**Activities**

## Lesson 1: Advanced Visualizations and Geospatial Data

- Waffle Charts & Word Cloud
- Seaborn and Regression Plots
- Hands-on Lab: Waffle Charts, Word Clouds, and Regression Plots



**Welcome to the Course**

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

**Introduction to Data Visualization**

- Practice Quiz: Advanced Visualization Tools

**Lesson 2: Visualizing Geospatial Data**

- Introduction to Folium
- Maps with Markers
- Choropleth Maps
- Hands-on Lab: Creating Maps and Visualizing Geospatial Data
- Practice Quiz: Visualizing Geospatial Data
- Module 3 Summary: Advanced Visualizations and Geospatial Data
- Module 3 Cheat Sheet
- Module 3 Graded Quiz: Advanced Visualizations and Geospatial Data

**Module 4****Title:** Creating Dashboards with Plotly and Dash



## Welcome to the Course

- Video:** Welcome to the Course  
3 min

- Reading:** How to Make the Most of this Course  
5 min

- Reading:** Syllabus  
10 min

## Introduction to Data Visualization

### Objectives

*By the end of this week, you will be able to:*

- Identify different web-based dashboarding tools available in Python
- Explore Plotly and its two sub-modules
- Use Plotly graph objects and Plotly express to create charts
- Discover Dash and its two components
- Describe the callback function
- Determine the process of connecting core and HTML components using callback

### Activities

Lesson 1: Creating Dashboards with Plotly and Dash

- Dashboarding Overview
- Additional Resources for Dashboards
- Introduction to Plotly
- Additional Resources for Plotly





## Welcome to the Course

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

## Introduction to Data Visualization

- Plotly Basics: Scatter, Line, Bar, Bubble, Histogram, Pie, Sunburst
- Practice Quiz: Creating Dashboards with Plotly

### Lesson 2: Working with Dash

- Introduction to Dash
- Overview of Cloud IDE lab environment
- Dash Basics: HTML and Core Components
- Additional Resources for Dash
- Make Dashboards Interactive
- Additional Resources for Interactive Dashboards
- Add Interactivity: User Inputs and Callbacks
- Understanding the Lab Environment
- Flight Delay Time Statistics Dashboard
- Practice Quiz: Working with Dash
- Module 4 Summary: Creating Dashboards with Plotly and Dash
- Module 4 Cheat Sheet



**Welcome to the Course**

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

**Introduction to Data Visualization**

- Module 4 Graded Quiz: Creating Dashboards with Plotly and Dash

## Module 5

**Title: Final Project and Exam****Description**

The primary focus of this module is to practice the skills gained earlier in the course and then demonstrate those skills in your final assignment. For the final assignment you will analyze historical automobile sales data covering periods of recession and non-recession. You will bring your analysis to life using visualization techniques and then display the plots and graphs on dashboards. Finally, you will submit your assignment for peer review and you will review an assignment from one of your peers. To wrap up the course you will take a final exam in the form of a timed quiz.

**Objectives**

*By the end of this week, you will be able to:*

- Practice visualization skills
- Practice creating a dashboard
- Create various visualizations using a number of plot libraries
- Create a dashboard and add interactivity





## Welcome to the Course

Video: Welcome to the Course

3 min

Reading: How to Make the Most of this Course

5 min

Reading: Syllabus

10 min

## Introduction to Data Visualization

- Review and grade an assignment submitted by peers

## Activities

### Lesson 1: Practice Project

- Practice Project Overview
- Practice Assignment: Part 1 - Analyzing wildfire data in Australia
- Practice Assignment: Part 2 - Creating Dashboards

### Lesson 2: Final Project

- Final Project Overview
- Final Assignment: Part 1 - Create Visualizations using Matplotlib, Seaborn & Folium
- Final Assignment: Part 2 - Create Dashboard with Plotly and Dash
- Final Assignment: Part 3 - Submission and Grading
- Final Exam: Data Visualization with Python - Timed Quiz

### Lesson 3: Course Wrap Up

- Course Summary
- Congratulations and Next Steps



# Overview of Data Visualization

---

© IBM Corporation. All rights reserved.

# What you will learn

---



Explain what Data Visualization is and why it is important



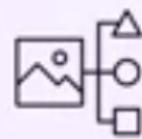
Describe the uses for Data Visualization



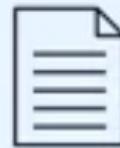
Explore the best practices for Data Visualization

# What is Data Visualization?

---



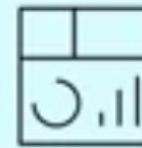
**Graphical representation** of data and information.



Can take a variety of forms.



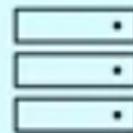
**Basic charts and graphs**



**Interactive dashboards and maps**

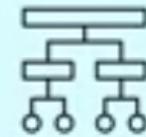
# Why build visuals?

- Easily Understand Data
  - Highlights patterns, trends, and relationships
- Communicate Insights
  - Simplifies the data for stakeholders



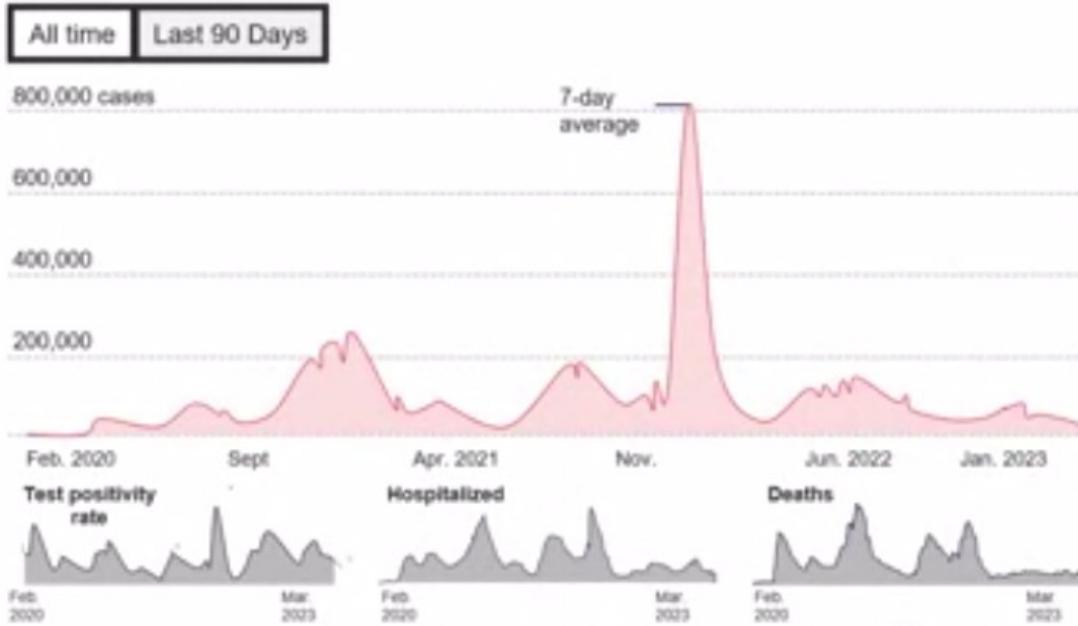
# Why build visuals?

- Identify trends and data patterns
  - Helps gain new insights
- Present data for better understanding and interpretation



# Leveraging the power of data visualization

## New reported cases



## Vaccinations

	FULLY VACCINATED	WITH A BOOSTER
All ages	68%	34%
65 and up	93%	67%

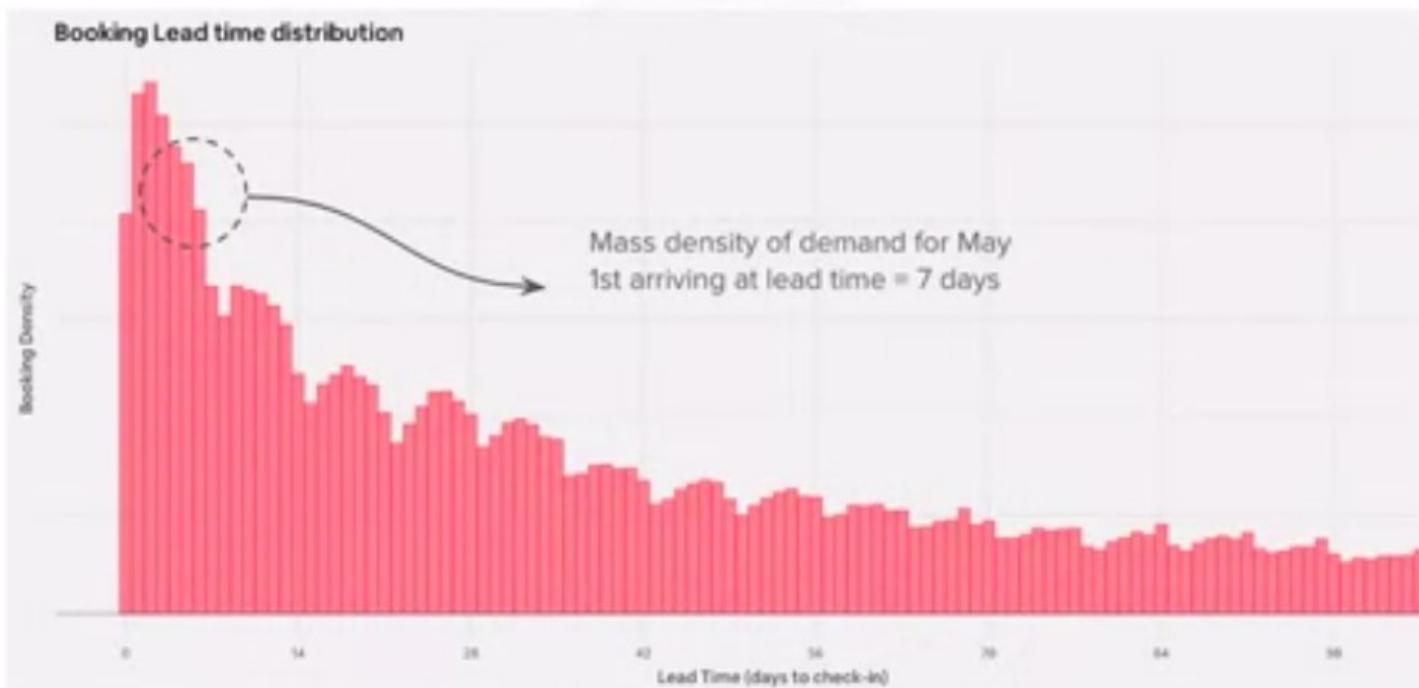
[See more details >](#)

[About this data](#)

Special offer: Subscribe for Rs. 4,500 - Rs. 600 for your first year.

SOURCE: <https://www.nytimes.com/interactive/2021/us/covid-cases.html>

# Leveraging the power of data visualization



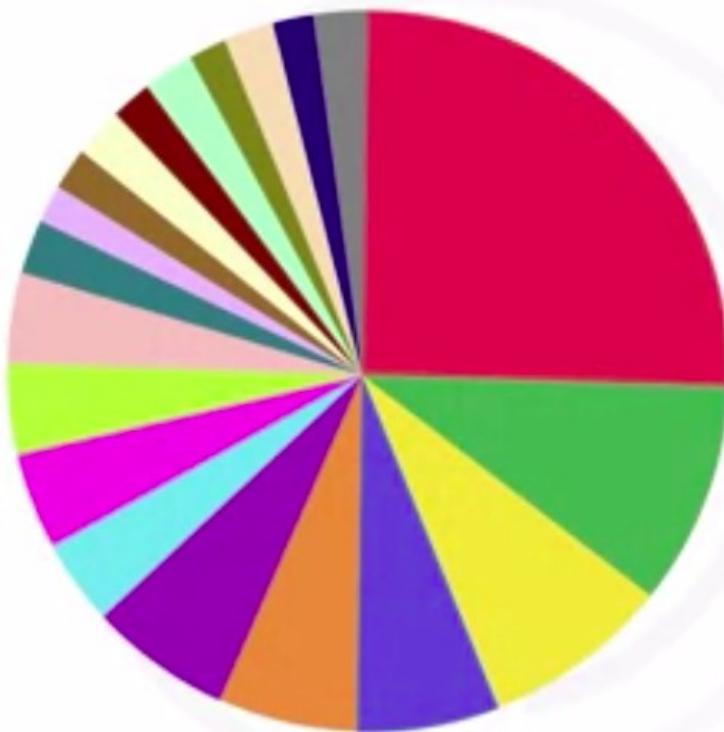
SOURCE: <https://medium.com/airbnb-engineering/learning-market-dynamics-for-optimal-pricing-97cffbcc53e3>

# Leveraging the power of data visualization

## Spotify Pie

Bake Your Monthly Genre Pie

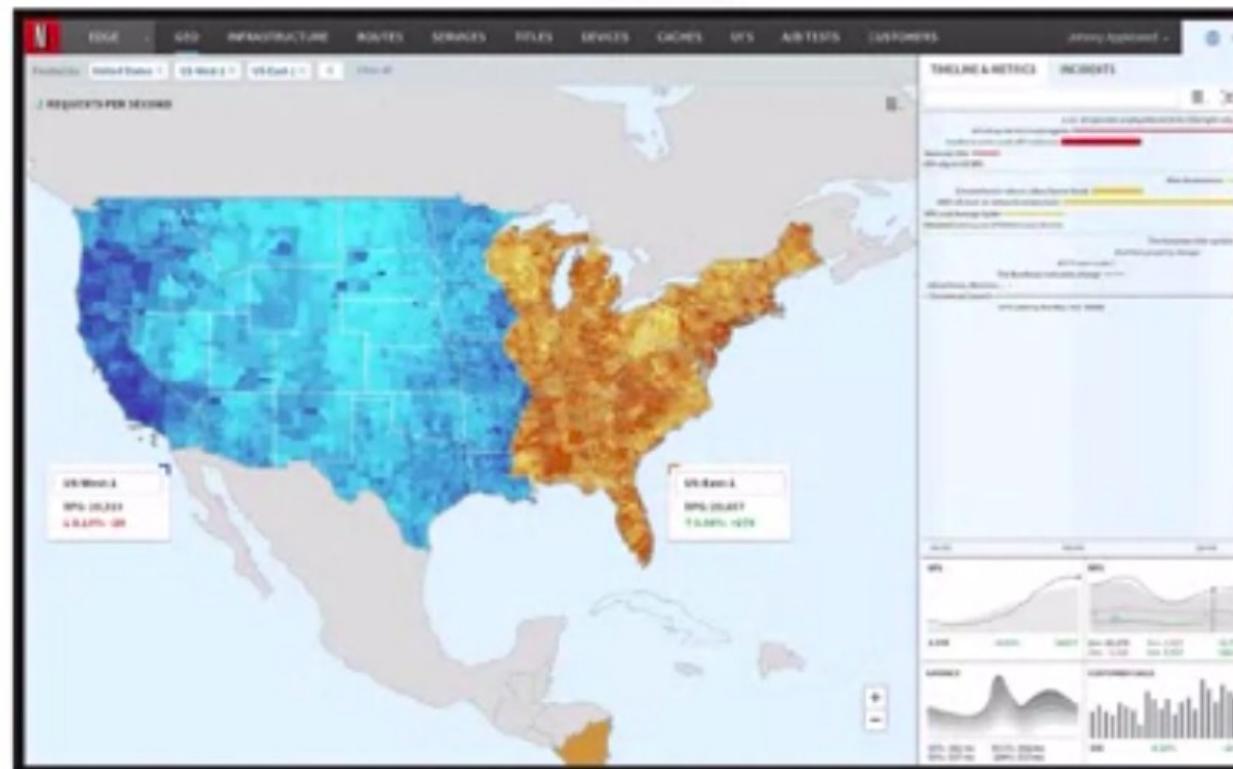
LOGIN TO SPOTIFY



Taylor Swift  
Harry Styles  
Florence + The Machine  
Phoebe Bridgers  
One Direction  
Lorde  
Charli XCX  
Niall Horan  
Jack Harlow  
ASAP Rocky  
Kacey Musgraves  
Tyler, The Creator  
Hozier  
Mitski  
Omar Apollo  
5 Seconds of Summer

SOURCE: <https://metro.co.uk/2023/04/11/what-is-spotify-pie-and-how-to-make-your-own-personalised-chart-18593008/>

# Leveraging the power of data visualization



SOURCE: <https://netflixtechblog.com/improving-netflixs-operational-visibility-with-real-time-insight-tools-ab5e7af062e5>

# Leveraging the power of data visualization

---

- Business and finance
- Healthcare
- Education
- Government
- Research and science
- Entertainment

# Best practices

---

- Data is accurately represented
- Message is clearly communicated



# Best practices

---

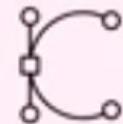
- Choose the right type of Visualization
  - Varied visualization techniques
- Keep it simple
  - Use of line chart and bar chart
- Use clear Labeling and Formatting
  - Including a clear title and legend



# Best practices

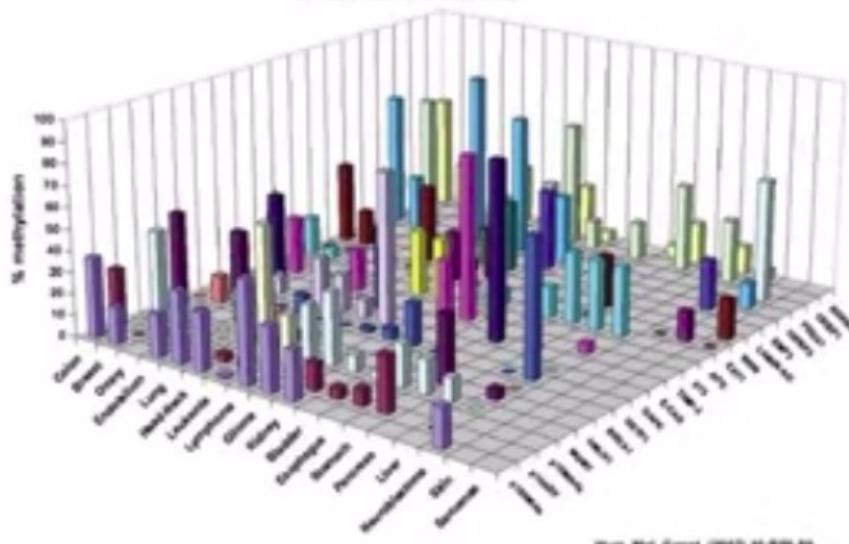
---

- Keep visualization focused
  - Use necessary data and labels only
- Consider the audience
  - Visualization should be tailored

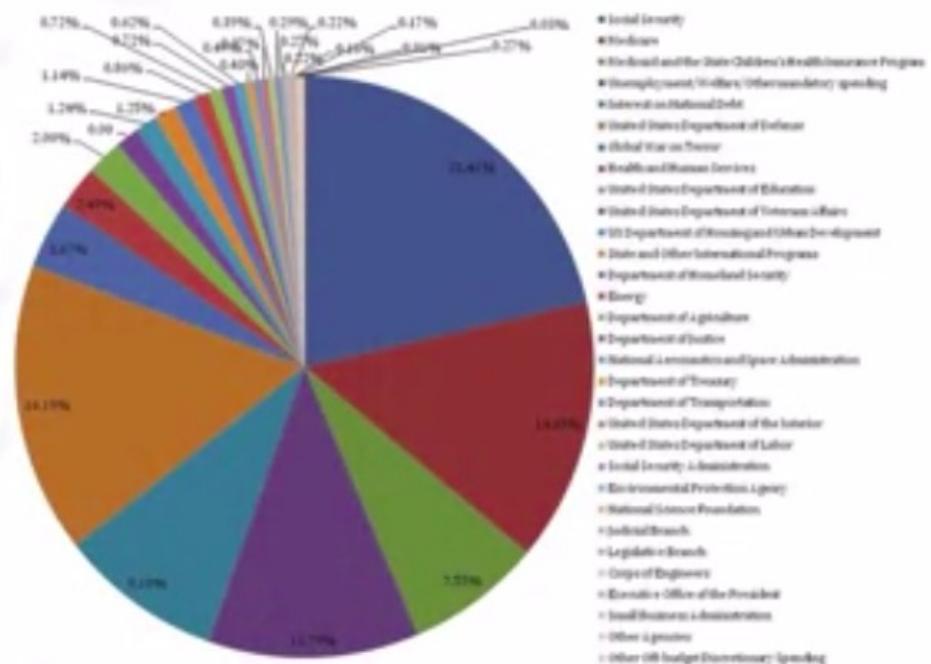


# Best practices (bad example)

A CpG Island Hypermethylation Profile of Human Cancer

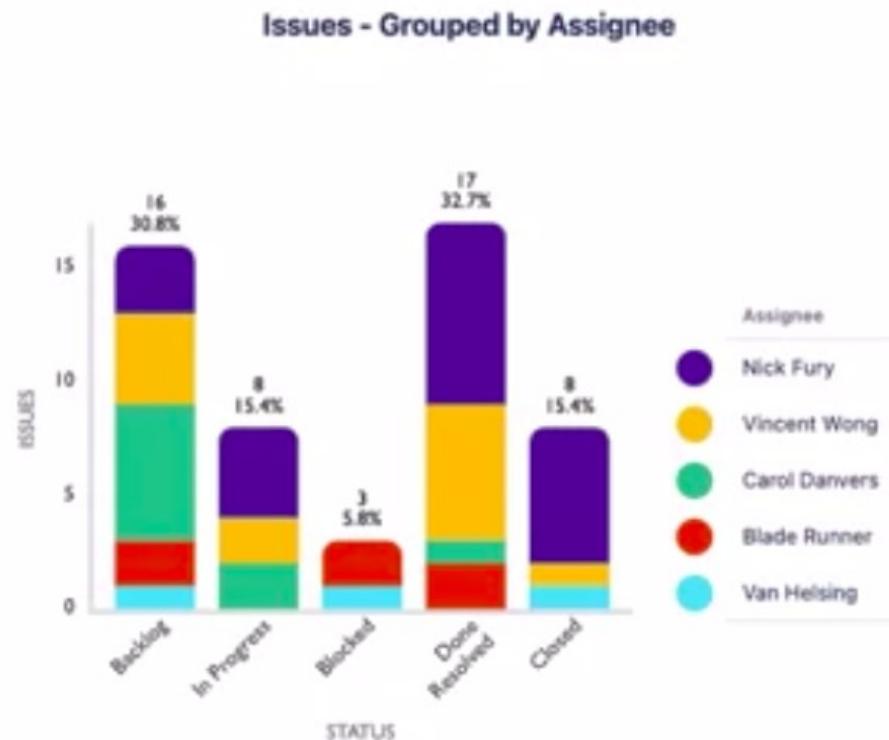


SOURCE: <https://www.livestories.com/blog/five-ways-to-fail-data-visualization>



SOURCE: [https://en.wikipedia.org/wiki/2007\\_United\\_States\\_federal\\_budget#](https://en.wikipedia.org/wiki/2007_United_States_federal_budget#)

# Best practices (good example)



SOURCE: <https://www.oldstreetsolutions.com/good-and-bad-data-visualization>

# DARKHORSE ANALYTICS

---

- Created in 2008
- Specializes in quantitative consulting



# DARKHORSE ANALYTICS

---

When creating a visual, always remember:

- Less is more effective
- Less is more attractive
- Less is more impactful

# Recap

---

In this video, you learned that:

- Data visualization is the process of presenting data in a visual format
- Data visualization has diverse use cases, such as in business, science, healthcare, and finance.
- It is important to follow data visualization best practices for presenting data

# Types of Plots

---

© IBM Corporation. All rights reserved.

# What you will learn

---



Recognize different types of plots available for visualizing data



Identify the characteristics and appropriate use cases for each type of plot

# Data visualization

---

- Represents data in visual formats
- Effectively communicates information, trends, and insights
- Uncovers patterns and identifies trends
- It presents complex information in an easier way to understand

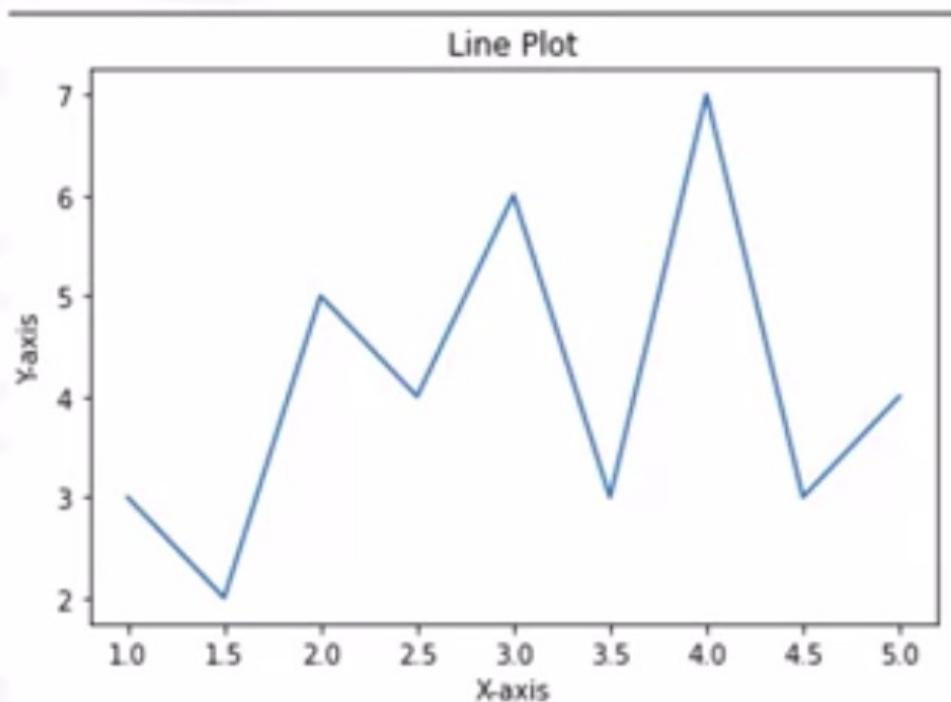
# Types of plots

Different types of plots:

- Line plot
- Bar plot
- Scatter plot
- Box plot
- Histogram

# Line plot

- Displays data as a series of data points connected by straight lines

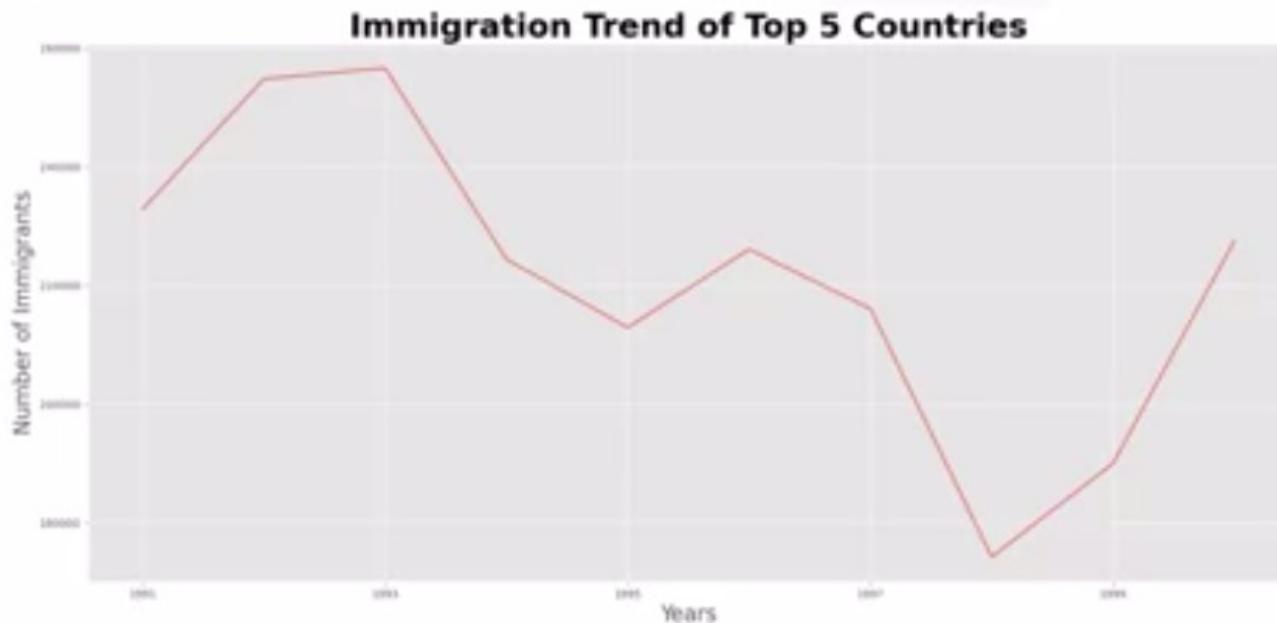


## Line plot: Use case

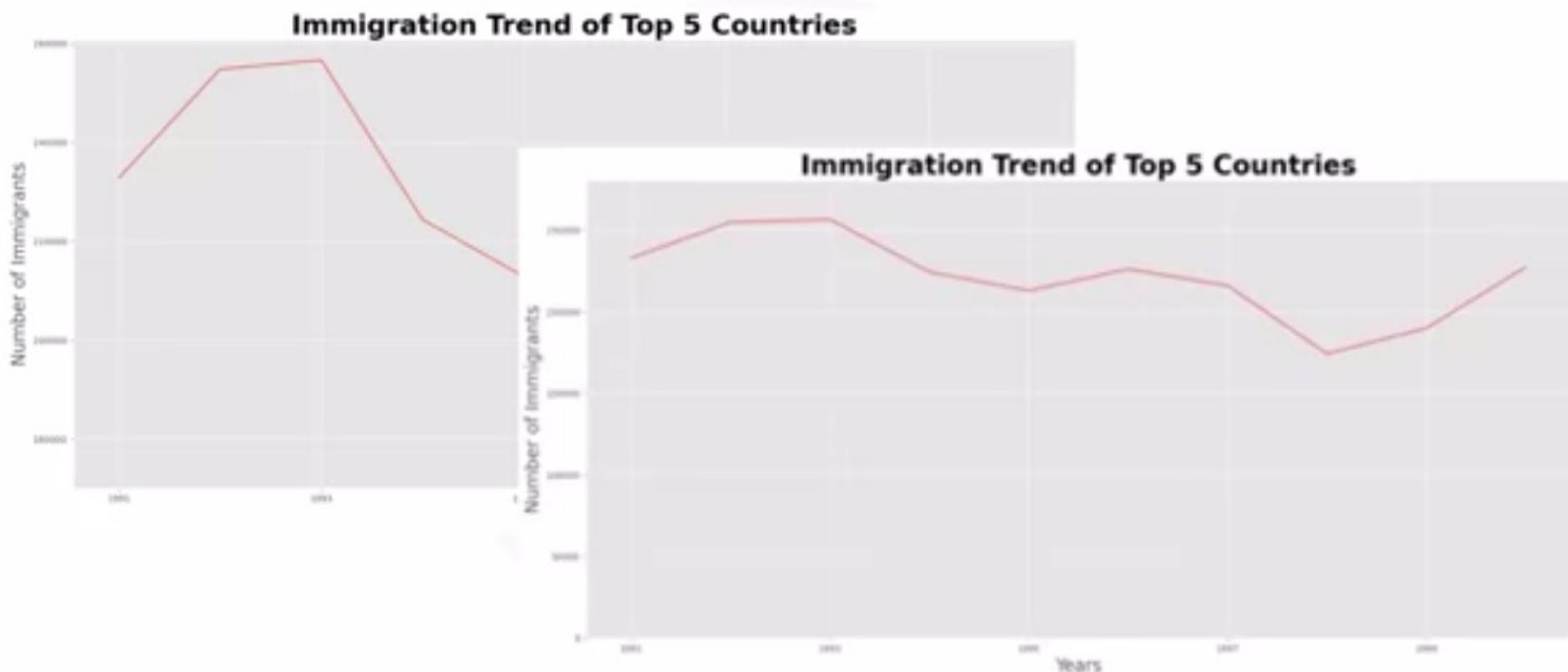
---

- Display trends over time
- Compare data sets with a continuous independent variable
- Illustrate cause-and-effect relationships.
- Visualize continuous data

# Line plot

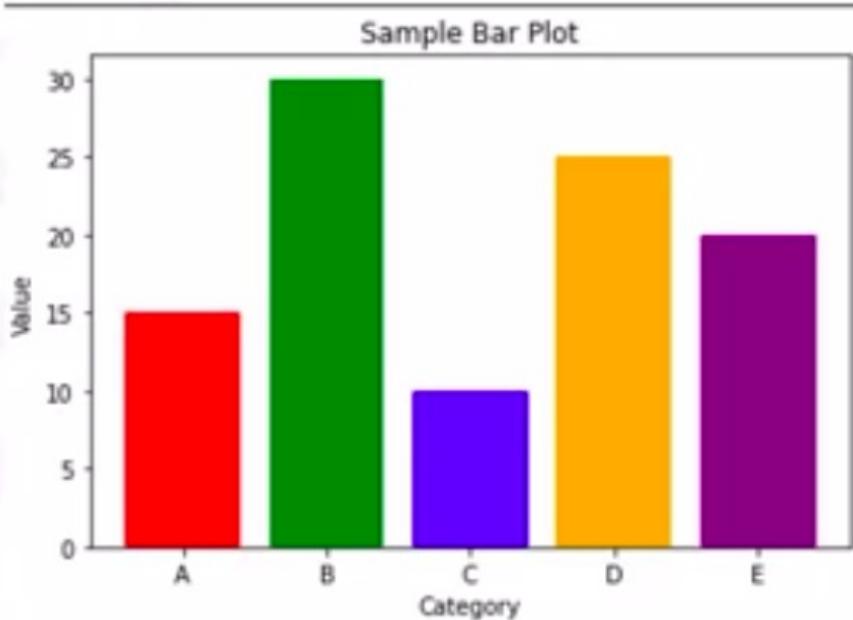


# Line plot



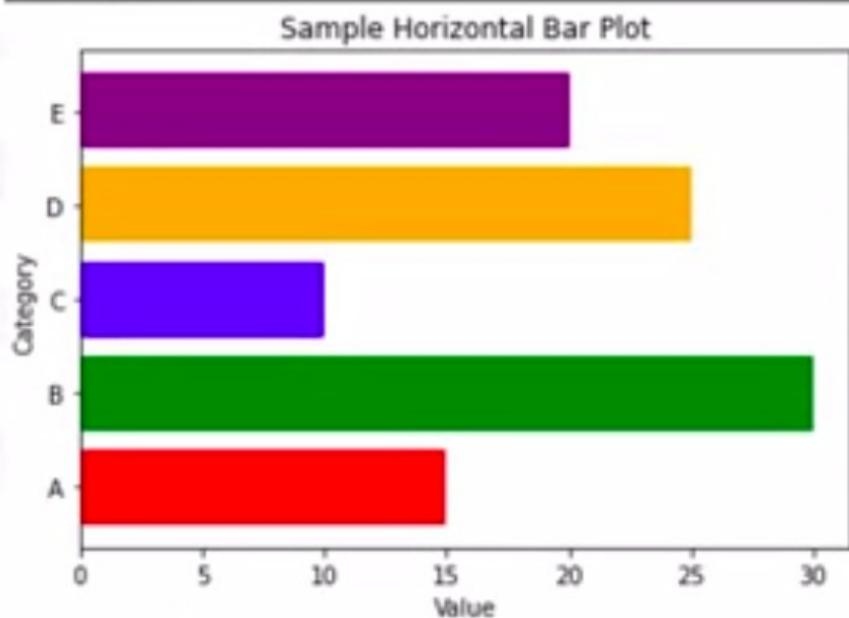
# Bar plot

- Displays data using rectangular bars
- The height or length of the bars represents the magnitude of the data



# Bar plot

- Displays data using rectangular bars
- The height or length of the bars represents the magnitude of the data
- Displays bars either vertically or horizontally

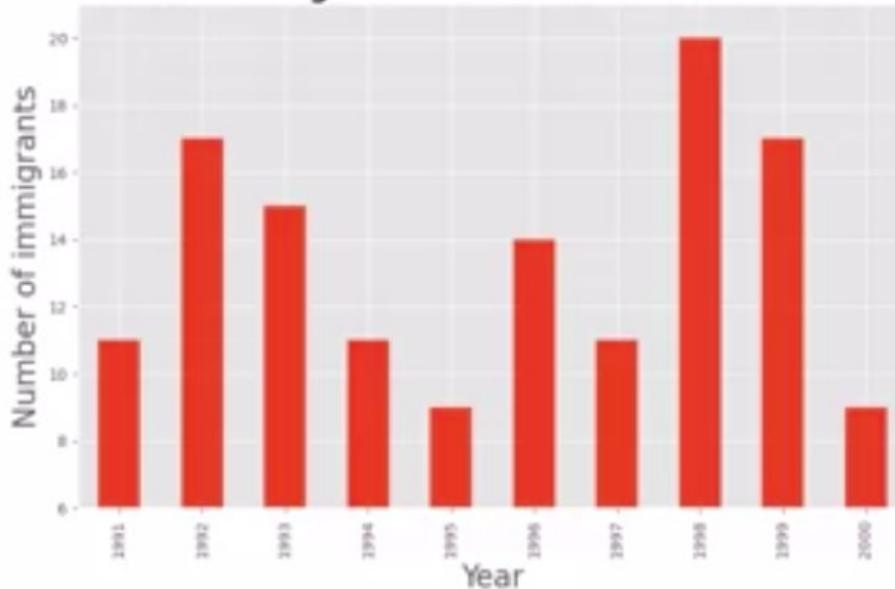


## Bar plot: Use cases

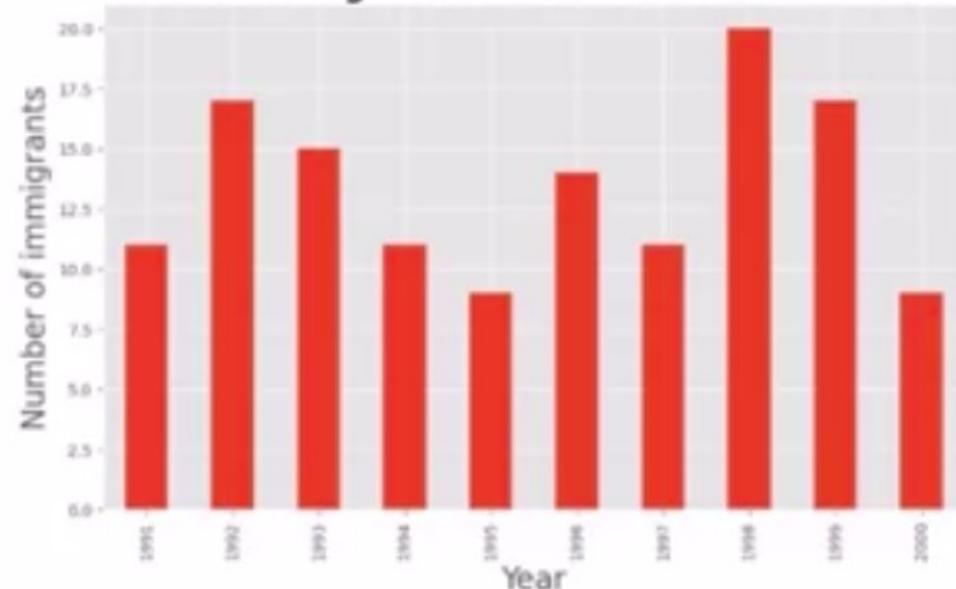
- Compare different categories or groups
- Display discrete data that has distinct categories
- Show how different categories contribute to a whole
- Visualize data that you can easily rank or ordered

# Bar plot

Icelandic immigrants to Canada from 1980 to 2013

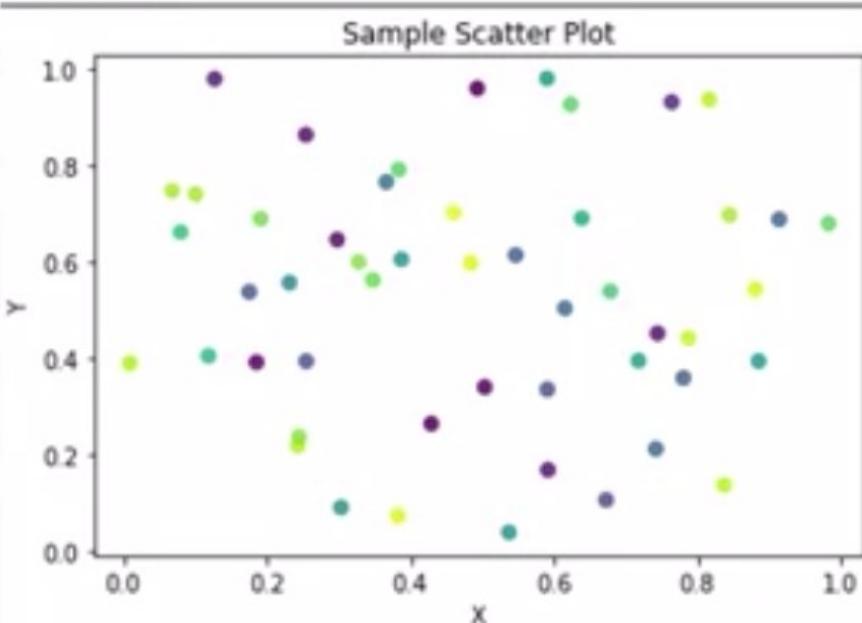


Icelandic immigrants to Canada from 1980 to 2013



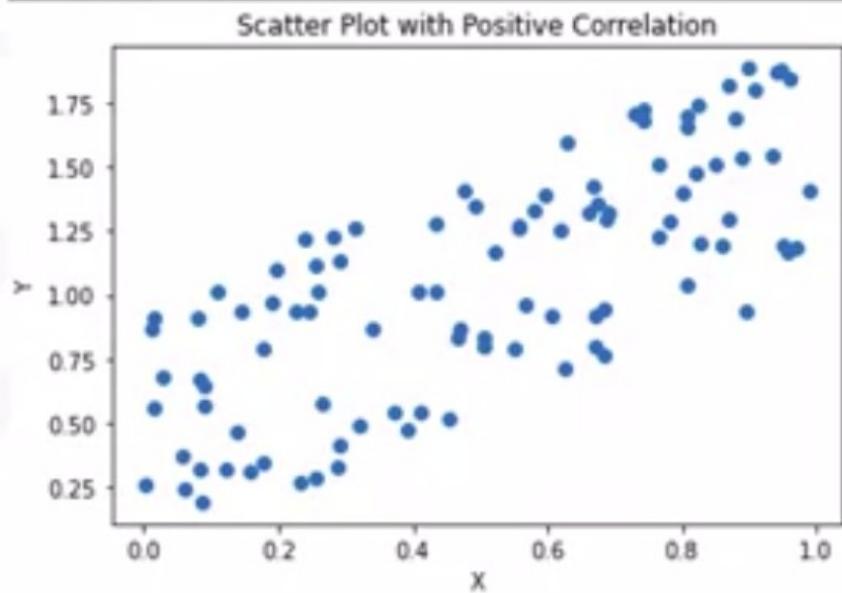
# Scatter plot

- Uses cartesian coordinates to display values for two variables
- Value of one variable determines the position on the horizontal axis, and the value of the other variable determines the position on the vertical axis



# Scatter plot: Use case

- Examine the relationship between two continuous variables

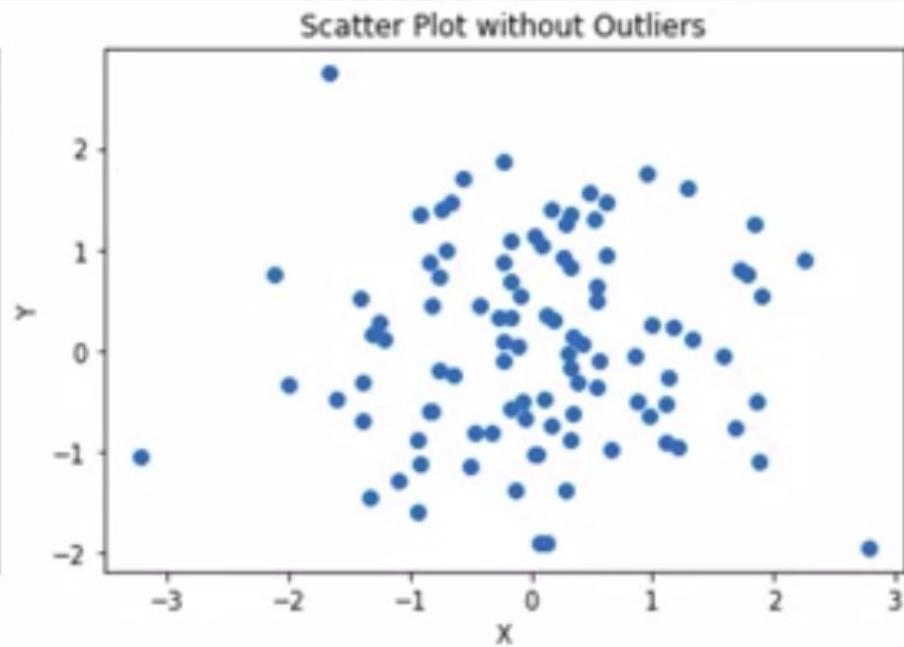
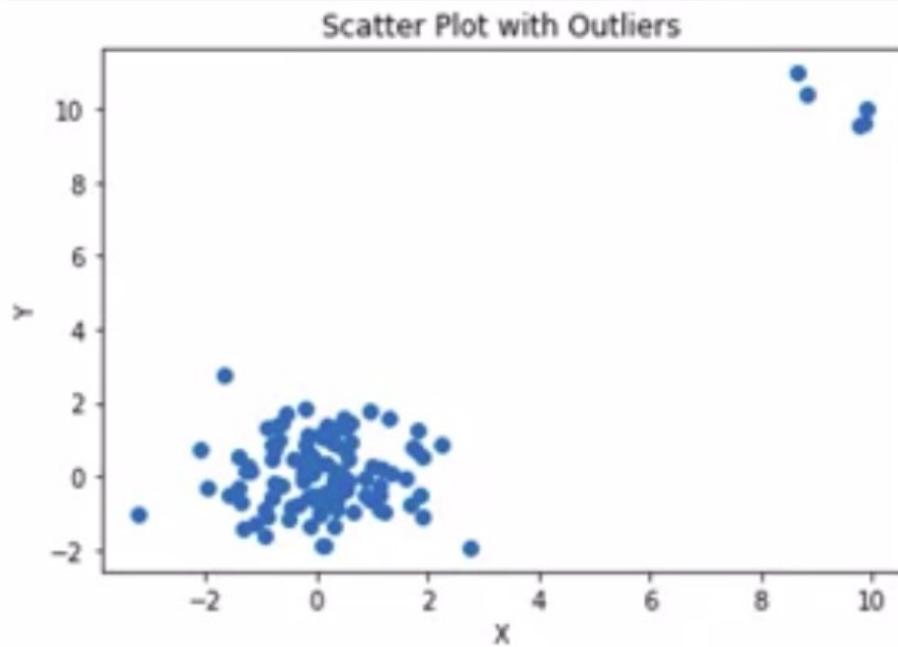


## Scatter plot: Use case

---

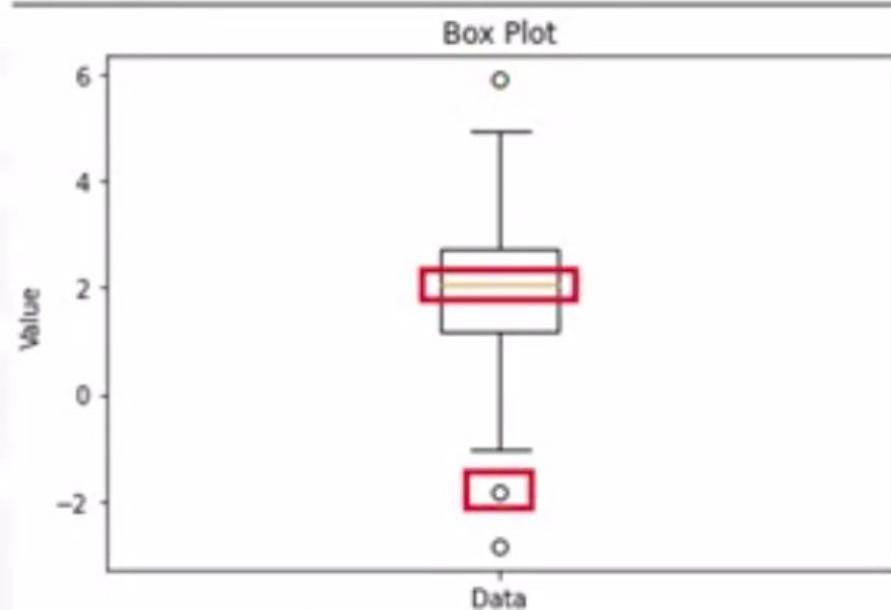
- Examine the relationship between two continuous variables
- Investigate patterns or trends in data
- Detect outliers or unusual observations
- Identify clusters or groups in the data
- Explore complex data

# Scatter plot



# Box plot

- Displays the distribution of a dataset along with key statistical measures
- Represents the interquartile range as a box, a median line, and whiskers indicating range
- Represents outliers as individual data points beyond whiskers



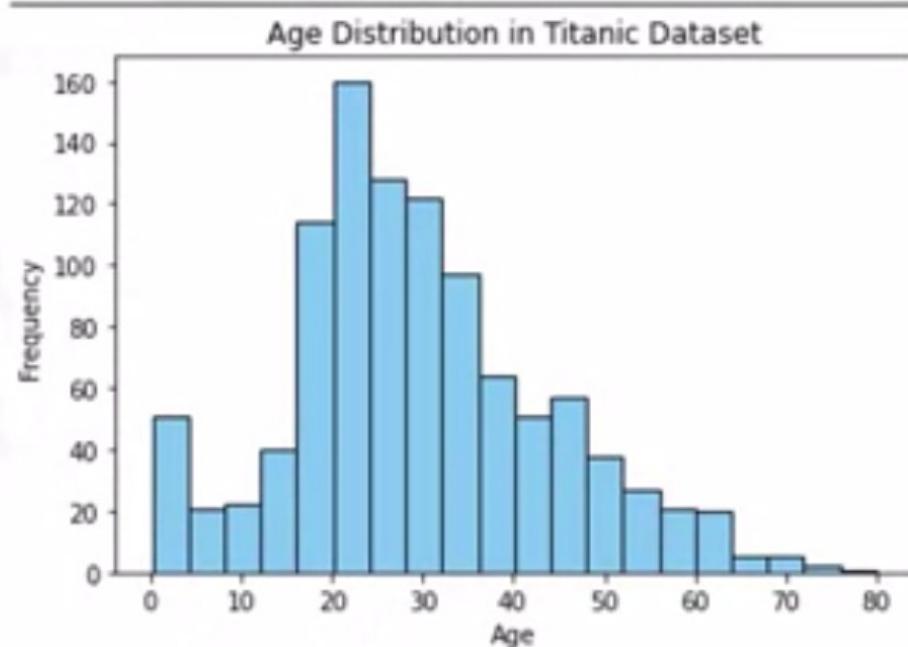
## Box plot: Use case

---

- Compare the distribution of a continuous variable across different categories or groups
- Examine spread and skewness of a dataset, visualizing quartiles
- Identify and analyze potential outliers
- Visualize summary statistics
- Compare distributions of multiple variables in datasets

# Histogram

- Graphical representation of dataset distribution
- Shows frequency or relative frequency within intervals
- Bars represent the data count in each interval

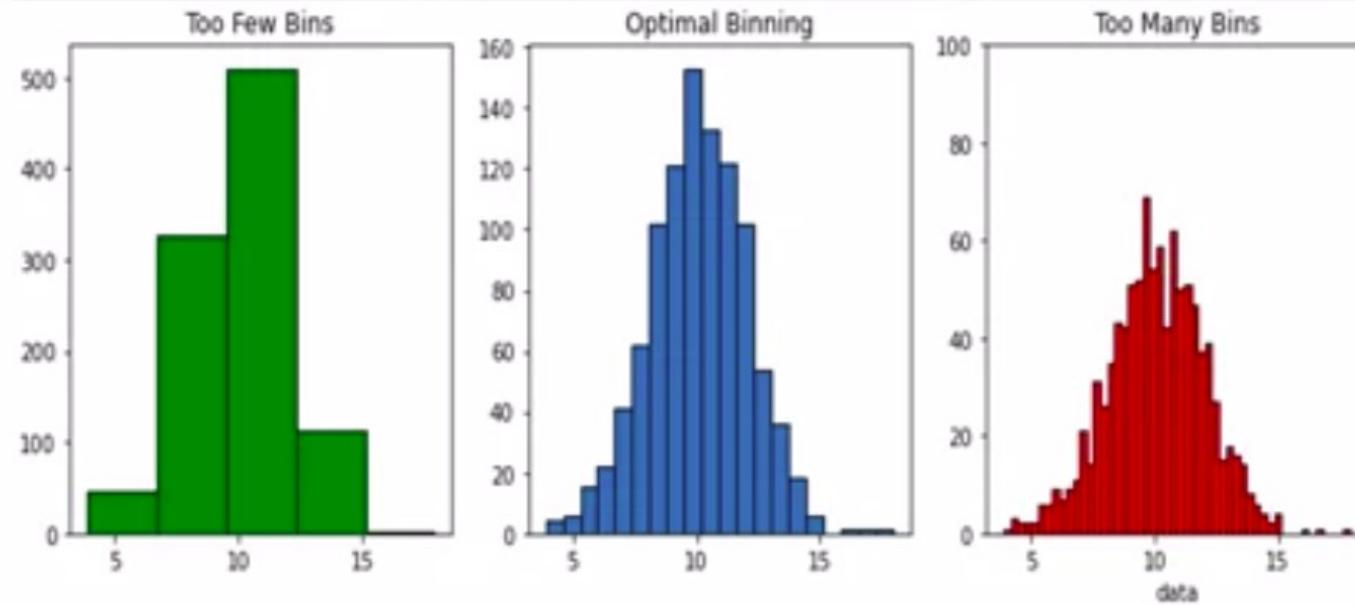


# Histogram: Use case

---

- Understand data distribution
- Visually depict the shape of the data
- Assess skewness in the data
- Showcase data variability and spread

# Histogram



# Recap

---

In this video, you learned that:

- There are various types of plots commonly used in data visualization.
- Line Plots: Shows trends and changes over time.
- Bar Plots: Compares categories or groups.
- Scatter Plots: Examines relationships between variables.
- Box Plots: Displays distribution and identifies outliers.
- Histograms: Illustrates data distribution within intervals.

# Plot Libraries

---

© IBM Corporation. All rights reserved.

# What you will learn

---



Explore some popular plot libraries for data visualization



Identify the features of plot libraries in Python

# Popular plot libraries

---

Plot libraries in Python:

- Matplotlib
- Pandas
- Seaborn
- Folium
- Plotly
- PyWaffle

# Matplotlib

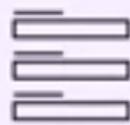
---



- General purpose plotting library
- Integrates well with libraries and frameworks

# Matplotlib

## Features:



Creates a wide variety of plots



Customizes various elements



Empowers data visualization tasks

# Pandas

---



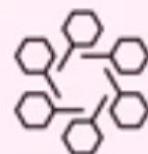
**pandas**

- Users employ it for data manipulation
- Its functions are built on Matplotlib

# Pandas

---

## Features



Integrates  
seamlessly with  
Pandas data  
structure



Analyzes  
exploratory data  
using visualization  
capabilities

# Seaborn

---



- Great option for specialized statistical visualizations
- It offers a variety of stylish plots

# Seaborn

---

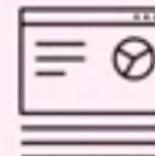
## Features



Offers various  
color palettes and  
styles



Offers functions to  
combine multiple  
plots in a grid  
layout



Integrates with  
Pandas

# Folium

---

## Folium



- Excellent option for geospatial data visualization
- Builds interactive and customizable maps

## Feature



Integrates with popular data analysis libraries

# Plotly

---



- Has highly interactive plots and dashboards
- Can create a variety of plots

# Plotly

---

## Features



Builds interactive  
dashboards



Enables plotting  
in a web browser

# PyWaffle

---

- Visualizes categorical data using waffle charts

## Features



Provides a unique  
way to represent  
proportions

# Recap

---

In this video, you learned that:

- Matplotlib is a plotting library that offers a wide range of plotting capabilities.
- Pandas is a plotting library that provides Integrated plotting functionalities for data analysis.
- Seaborn is a specialized library for statistical visualizations, offering attractive default aesthetics and color palettes.
- Folium is a Python library that allows you to create interactive and customizable maps.
- Plotly is an interactive and dynamic library for data visualization that supports a wide range of plot types and interactive features.
- PyWaffle enables you to visualize proportional representation using squares or rectangles.



# Introduction to Matplotlib

---

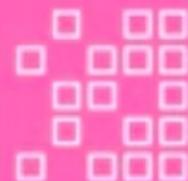
© IBM Corporation. All rights reserved.

# What you will learn

---

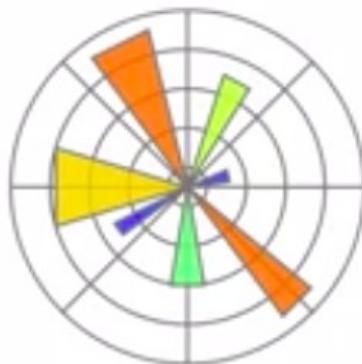


Explain what  
Matplotlib is and why  
it was created

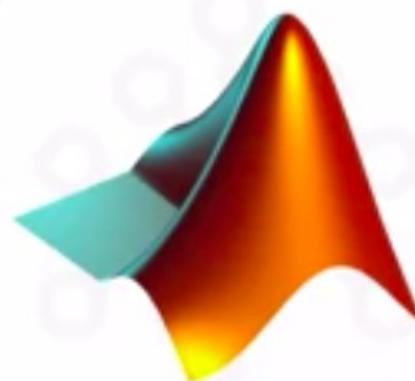


Describe the uses for  
Matplotlib

# Matplotlib - history



John Hunter (1968 – 2012)



Matlab-based version of  
a proprietary software



EEG/ECOG Visualization Tool

# Matplotlib architecture



# Backend layer

---

Has three built-in abstract interface classes:

1. FigureCanvas: **matplotlib.backend\_bases.FigureCanvas**
  - Encompasses the area onto which the figure is drawn
2. Renderer: **matplotlib.backend\_bases.Renderer**
  - Knows how to draw on the FigureCanvas
3. Event: **matplotlib.backend\_bases.Event**
  - Handles user inputs such as keyboard strokes and mouse clicks

# Artist layer

---

- Comprised of one main object – **Artist**:
  - Knows how to use the Renderer to draw on the canvas.
- Title, lines, tick labels, and images, all correspond to individual **Artist** instances.
- Two types of **Artist** objects:
  1. **Primitive**: Line2D, Rectangle, Circle, and Text.
  2. **Composite**: Axis, Tick, Axes, and Figure.
- Each *composite* artist may contain other *composite* artists as well as *primitive* artists.

# Scripting layer

---

It is comprised mainly of Pyplot, a scripting interface lighter than the Artist layer. Let's see how we can generate the histogram of 100 random values using the Pyplot interface.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.randn(10000)
plt.hist(x, 100)
plt.title(r'Normal distribution with $\mu=0, \sigma=1$')
plt.savefig('matplotlib_histogram.png')
plt.show()
```

# Scripting layer

---

It is comprised mainly of Pyplot, a scripting interface lighter than the Artist layer. Let's see how we can generate the histogram of 100 random values using the Pyplot interface.

```
import matplotlib.pyplot as plt
import numpy as np

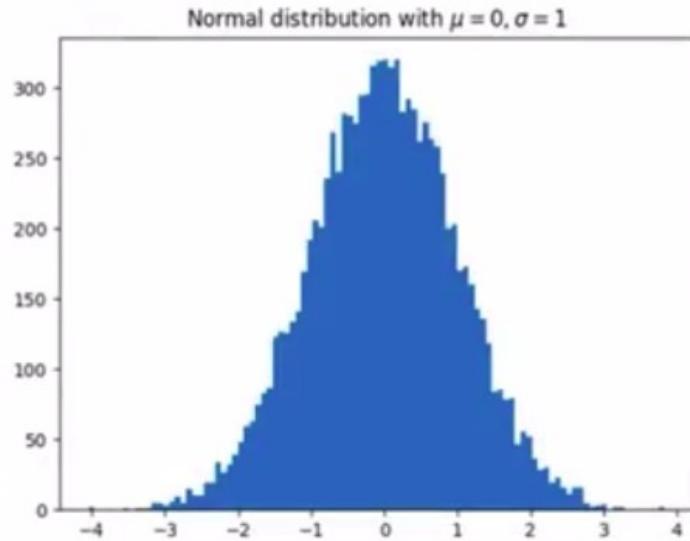
x = np.random.randn(10000)
plt.hist(x, 100)
plt.title(r'Normal distribution with $\mu=0, \sigma=1$')
plt.savefig('matplotlib_histogram.png')
plt.show()
```

# Scripting layer

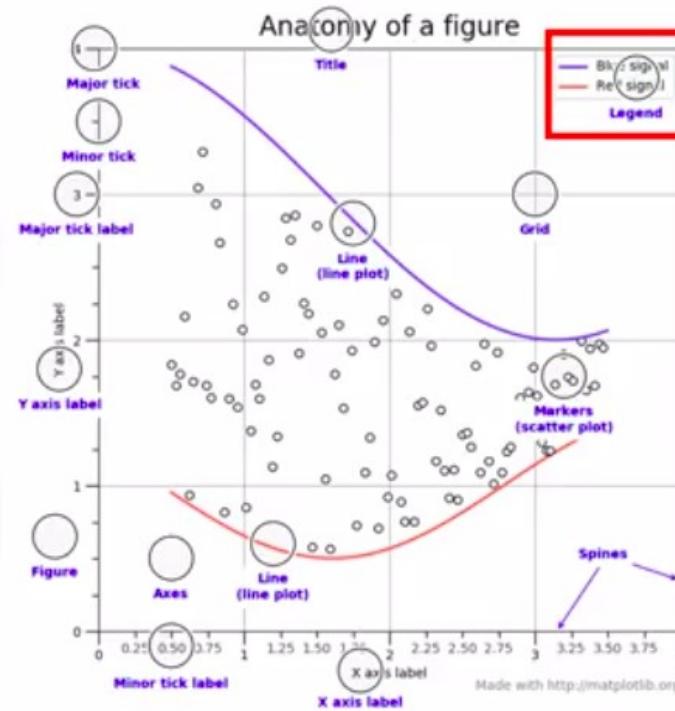
It is comprised mainly of Pyplot, a scripting interface lighter than the Artist layer. Let's see how we can generate the histogram of 100 random values using the Pyplot interface.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.randn(10000)
plt.hist(x, 100)
plt.title(r'Normal distribution with $\mu=0, \sigma=1$')
plt.savefig('matplotlib_histogram.png')
plt.show()
```



# Components of a plot



SOURCE: <https://matplotlib.org/stable/gallery/showcase/anatomy.html>

## Further reading

You can find more information about Matplotlib: its history and architecture, by following the link below:

<http://www.aosabook.org/en/matplotlib.html>

# Recap

---

In this video, you learned that:

- Matplotlib is one of the most widely used data visualization libraries in Python.
- Matplotlib was initially developed as an EEG/ECoG visualization tool.
- Matplotlib's architecture is composed of three main layers: Backend layer, Artist layer, and the Scripting layer.
- The anatomy of a plot refers to the different components and elements that make up a visual representation of data.

# Basic Plotting with Matplotlib

---

© IBM Corporation. All rights reserved.

# What you will learn

---



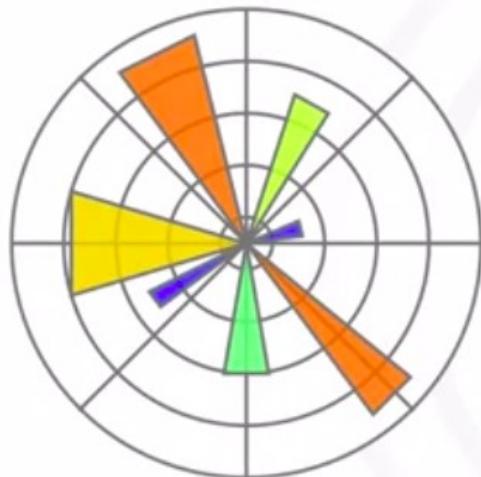
Explore how to use Matplotlib to create plots by employing Jupyter Notebook



Create conventional visualization tools using the plot function

# Matplotlib – Jupyter Notebook

---

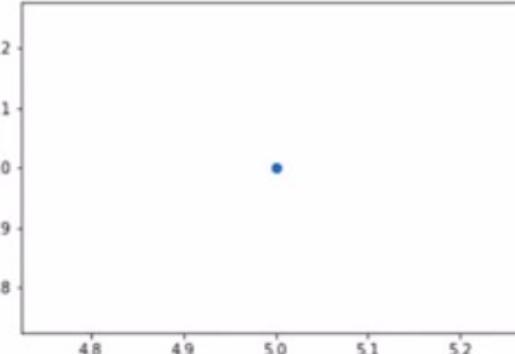


# Matplotlib Backends – Inline

jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 2 hours ago (autosaved) Logout Python 2

In [1]: `%matplotlib inline  
import matplotlib.pyplot as plt`

In [2]: `plt.plot(5, 5, 'o')  
plt.show()`

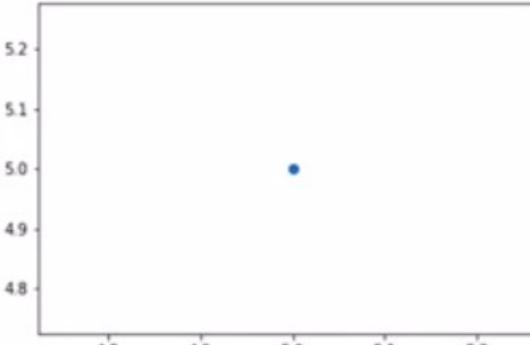


# Matplotlib Backends – Inline

jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 2 hours ago (autosaved) Logout Python 2

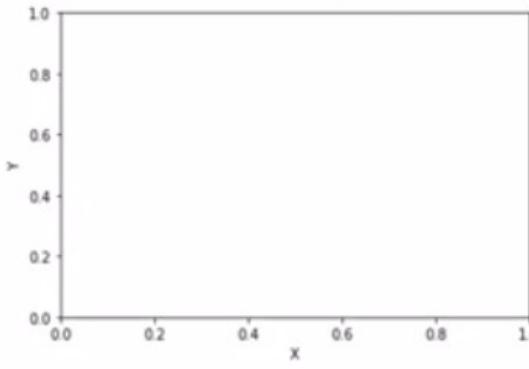
In [1]: `matplotlib inline  
import matplotlib.pyplot as plt`

In [2]: `plt.plot(5, 5, 'o')  
plt.show()`



In [3]: `plt.ylabel("Y")  
plt.xlabel("X")`

Out[3]: <matplotlib.text.Text at 0x110480810>



# Matplotlib Backends – Inline

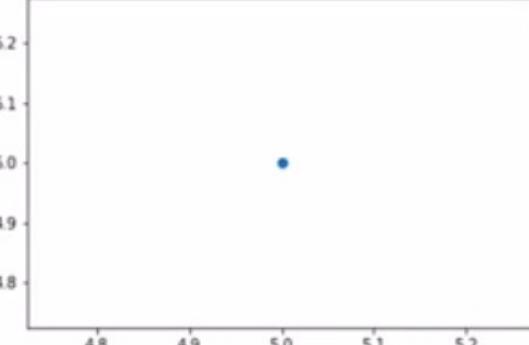
jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 2 hours ago (autosaved) Logout Python 2

In [1]:

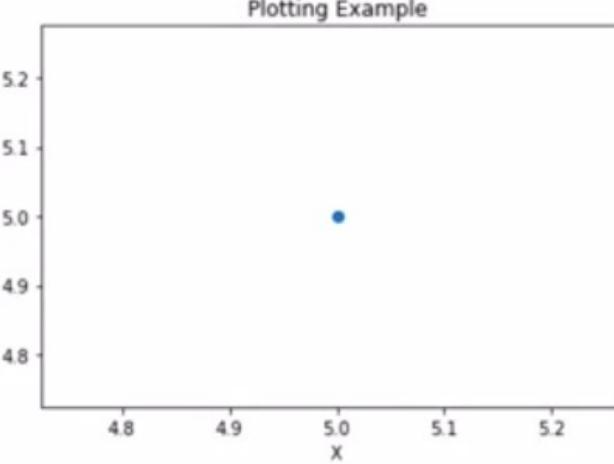
```
matplotlib inline
import matplotlib.pyplot as plt
```

In [2]:

```
plt.plot(5, 5, 'o')
plt.show()
```



plt.plot(5, 5, 'o')  
plt.ylabel("Y")  
plt.xlabel("X")  
plt.title("Plotting Example")  
plt.show()



# Matplotlib Backends – Notebook

jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 16 minutes ago (autosaved) Logout Python 2

In [1]: `matplotlib notebook  
import matplotlib.pyplot as plt`

In [2]: `plt.plot(5, 5, 'o')`

Figure 1

Out[2]: [`<matplotlib.lines.Line2D at 0x10784c790>`]

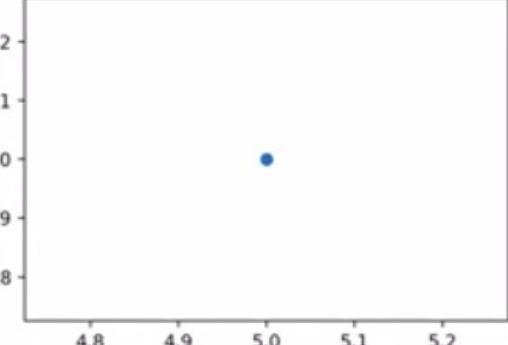
# Matplotlib Backends – Notebook

jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 16 minutes ago (autosaved) Logout Python 2

In [1]: `matplotlib notebook`  
`import matplotlib.pyplot as plt`

In [2]: `plt.plot(5, 5, 'o')`

Figure 1



Out[2]: [`<matplotlib.lines.Line2D at 0x10784c790>`]

# Matplotlib Backends – Notebook

Jupyter Matplotlib - Jupyter Notebook Last Checkpoint: 19 minutes ago (unsaved changes) Logout Python 2

File Edit View Insert Cell Kernel Help

In [1]: `matplotlib notebook  
import matplotlib.pyplot as plt`

In [2]: `plt.plot(5, 5, 'o')`

Figure 1

Plotting Example

In [3]: `plt.ylabel("Y")  
plt.xlabel("X")  
plt.title("Plotting Example")`

Out[3]: <matplotlib.text.Text at 0x1077f2910>

Out[2]: [<matplotlib.lines.Line2D at 0x10784c790>]

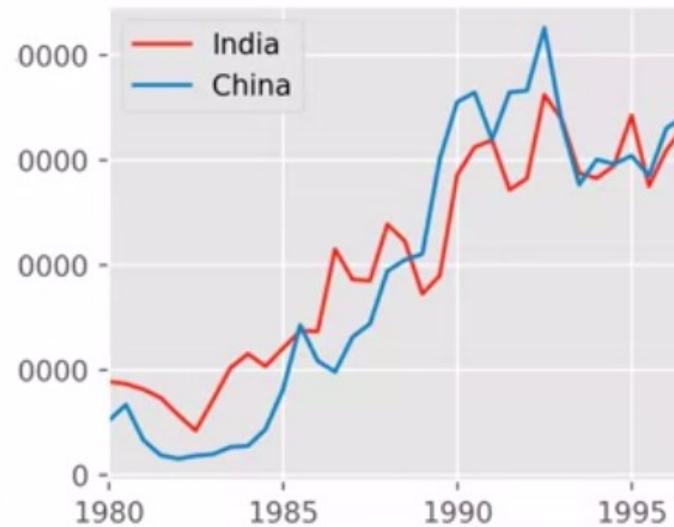
# Matplotlib – Pandas

india\_china\_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df.plot(kind="line")
```

Figure 1



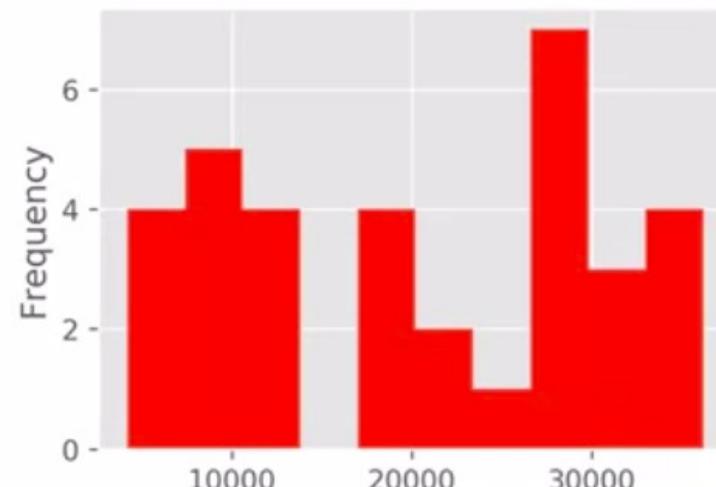
# Matplotlib – Pandas

india\_china\_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df["India"].plot(kind="hist")
```

Figure 1



# Recap

---

In this video, you learned that:

- Matplotlib is an established data visualization library that can be integrated in different environments.
- Jupyter Notebook is an open-source web application that allows you to create and share documents.
- Matplotlib has a number of different backends available.
- Adding labels and title to the plot is easy by using plt.

# Dataset on Immigration to Canada

---

© IBM Corporation. All rights reserved.

# What you will learn

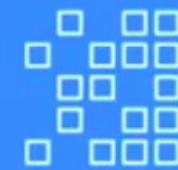
---



Understand the dataset to be used in this course for data visualization



Import data with Pandas as a DataFrame in your program



Process data to make it suitable for plotting

# Dataset

---

- The Population Division of the United Nations compiled data pertaining to 45 countries.
- For each country, annual data on the flows of international migrants is reported in addition to other metadata.
- We will primarily work with United Nations data on immigration to Canada.

# Immigration data to Canada



United Nations  
Population Division  
Department of Economic and Social Affairs

## *International Migration Flows to and from Selected Countries: The 2015 Revision*

POP/DB/MIG/Flow/Rev.2015

December 2015 - Copyright © 2015 by United Nations. All rights reserved

Suggested citation: United Nations, Department of Economic and Social Affairs, Population Division (2015).

International Migration Flows to and from Selected Countries: The 2015 Revision. (United Nations database,

Reporting country: Canada

Criterion: Citizenship

Classification		Origin/Destination		Major area		Region		Development region							
Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982	1983	1984	1985	
Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	39	39	47	71	340	
Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	0	0	0	0	0	
Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	67	71	69	63	44	
Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	1	0	0	0	0	
Immigrants	Foreigners	Andorra	one	Europe	925	Southern Europe	901	Developed regions	0	0	0	0	0	0	

# Read data into Pandas DataFrame

```
import numpy as np # useful for many scientific computing in Python
import pandas as pd # primary data structure library
from __future__ import print_function # adds compatibility to python 2
```

```
#install openpyxl
!pip install openpyxl
print('openpyxl installed')
```

```
df_can = pd.read_excel(
    'https://ibm.box.com/shared/static/lw190pt9zpy5bd1ptyg2aw15awomz9pu.xlsx',
    sheetname="Canada by Citizenship",
    skiprows=range(20),
    skip_footer = 2)
```

# Display DataFrame

```
df_can.head()
```

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	...	2004	2005	2006	2007	2008	2009	2010	2011
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	...	2978	3436	3009	2652	2111	1746	1758	2203
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	...	1450	1223	856	702	560	716	561	539
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	...	3616	3626	4807	3623	4005	5393	4752	4325
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	...	0	0	1	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	...	0	0	1	1	0	0	0	0

# Dataset - processed

	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
<b>Country</b>																					
Afghanistan	Asia	Southern Asia	Developing regions	16	39	39	47	71	340	496	...	3436	3009	2652	2111	1746	1758	2203	2635	2004	58639
Albania	Europe	Southern Europe	Developed regions	1	0	0	0	0	0	1	...	1223	856	702	560	716	561	539	620	603	15699
Algeria	Africa	Northern Africa	Developing regions	80	67	71	69	63	44	69	...	3626	4807	3623	4005	5393	4752	4325	3774	4331	69439
American Samoa	Oceania	Polynesia	Developing regions	0	1	0	0	0	0	0	...	0	1	0	0	0	0	0	0	6	
Andorra	Europe	Southern Europe	Developed regions	0	0	0	0	0	0	2	...	0	1	1	0	0	0	0	1	1	15

df\_canada

# Recap

---

In this video, you learned that:

- The population division of the United Nations compiled immigration data pertaining to 45 countries.
- The UN data on immigration to Canada shows data related to the number of people migrated.
- In order to start creating different types of plots of the data, you will need to import the data into a Pandas DataFrame.

# Line Plots

---

© IBM Corporation. All rights reserved.

# What you will learn

---



Describe line plot  
and its function



Determine when to  
use a line plot



Create a line plot  
from data in the  
dataset

# Line plots

---

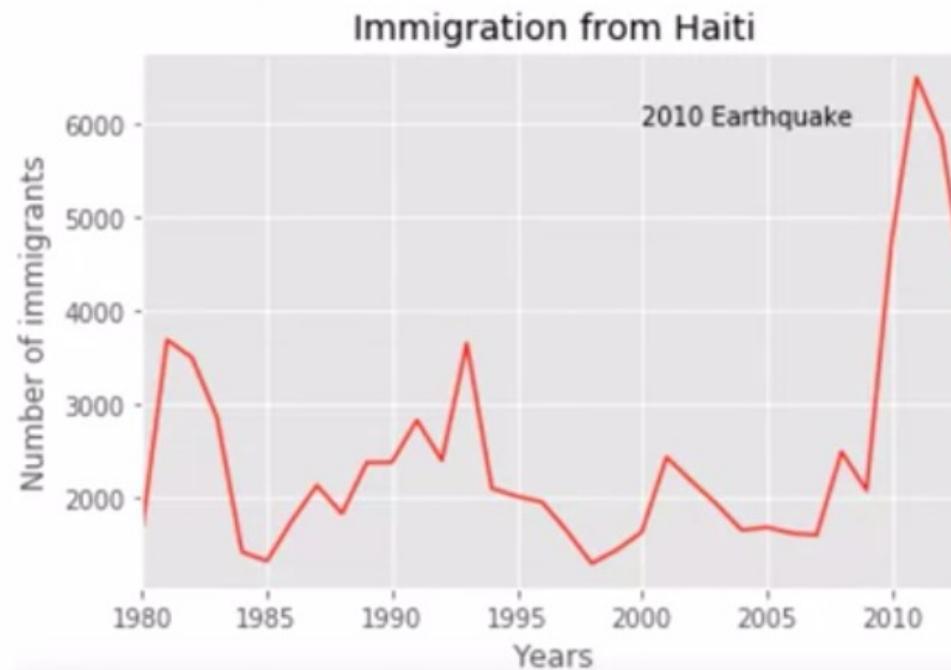
A line plot is a type of plot that displays information as a series of data points connected by straight lines.

1. Visualizing trends and changes over time
2. Showing relationships
3. Compare multiple data series
4. Highlighting sudden changes or anomalies

# Line plots

---

Spike of immigration from Haiti to Canada in 2010 due to earthquake



# Dataset - recap

---

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	...	2004	2005	2006	2007	2008	2009	2010	2011
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	...	2978	3436	3009	2652	2111	1746	1758	2203
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	...	1450	1223	856	702	560	716	561	539
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	...	3616	3626	4807	3623	4005	5393	4752	4325
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	...	0	0	1	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	...	0	0	1	1	0	0	0	0

# Dataset - processed

	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Country																					
Afghanistan	Asia	Southern Asia	Developing regions	16	39	39	47	71	340	496	...	3436	3009	2652	2111	1746	1758	2203	2635	2004	58639
Albania	Europe	Southern Europe	Developed regions	1	0	0	0	0	0	1	...	1223	856	702	560	716	561	539	620	603	15699
Algeria	Africa	Northern Africa	Developing regions	80	67	71	69	63	44	69	...	3626	4807	3623	4005	5393	4752	4325	3774	4331	69439
American Samoa	Oceania	Polynesia	Developing regions	0	1	0	0	0	0	0	...	0	1	0	0	0	0	0	0	6	
Andorra	Europe	Southern Europe	Developed regions	0	0	0	0	0	0	2	...	0	1	1	0	0	0	0	1	1	15

df\_canada

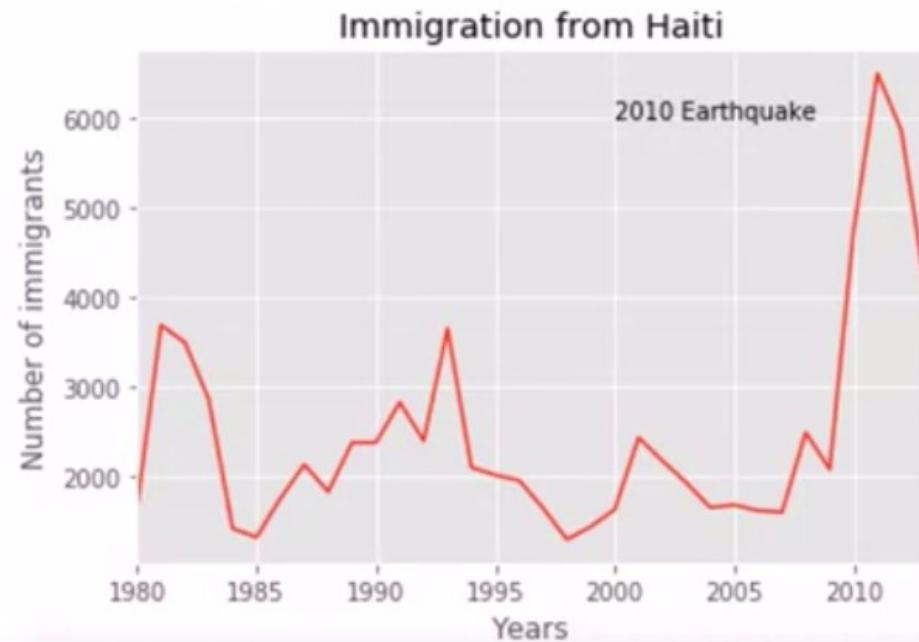
# Creating line plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt

years = list(map(str, range(1980, 2014)))

df_canada.loc['Haiti', years].plot(kind = 'line')
plt.title('Immigration from Haiti')
plt.ylabel('Number of immigrants')
plt.xlabel('Years')

plt.show()
```



# Recap

---

In this video, you learned that:

- A line plot is a plot in the form of a series of data points connected by straight line segments.
- Line plot is one of the most basic type of chart and is common in many fields.
- You can generate a line plot by assigning "line" to 'Kind' parameter in the plot() function.

Back Practice Quiz: Introduction to Data Visualization Tools  
Practice Quiz • 10 min • 5 total points

## Congratulations! You passed!

Grade received 100% To pass 80% or higher

Go to next item

1. Matplotlib was created by:

1 / 1 point

- James Gosling, a Canadian computer scientist.
- Cleve Moler, an American mathematician and computer programmer.
- Daniel Johnson, a German physicist.
- John Hunter, an American neurobiologist.

Correct

Correct! The creator of Matplotlib was John Hunter, an American neurobiologist.

2. True or False. Jupyter Notebook is an open-source web application that allows you to create and share documents.

1 / 1 point

- True

[Back](#) Practice Quiz: Introduction to Data Visualization Tools

Practice Quiz • 10 min • 5 total points

2. True or False. Jupyter Notebook is an open-source web application that allows you to create and share documents.

1 / 1 point

True

False

 Correct

Correct! Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, visualizations, and some explanatory text.

3. What is the code for the Matplotlib magic function?

1 / 1 point

'%matplotlib'

(%matplotlib)

{%matplotlib}

%matplotlib

 Correct

Correct! The magic function code is %matplotlib. If the plot gets generated in a new pop window, you can enforce generating plots into a browser using this code.

[Back](#)

## Practice Quiz: Introduction to Data Visualization Tools

Practice Quiz • 10 min • 5 total points

4. Fill in the blank. A line plot is a series of \_\_\_\_\_ points connected by straight line segments.

1 / 1 point

- Data
- Matplotlib
- Plotly
- Connection

 **Correct**

Correct! A line plot is a plot in the form of a series of data points connected by straight-line segments. A line plot is one of the most basic chart types common in many fields.

5. What is the most widely used data visualization library in Python?

1 / 1 point

- Plotly
- Matplotlib
- Jupyter Notebook

← Back Practice Quiz: Introduction to Data Visualization Tools

Practice Quiz • 10 min • 5 total points

Plotly

Connection

 Correct

Correct! A line plot is a plot in the form of a series of data points connected by straight-line segments. A line plot is one of the most basic chart types common in many fields.

5. What is the most widely used data visualization library in Python?

1 / 1 point

Plotly

Matplotlib

Jupyter Notebook

Pyplot

 Correct

Correct! Matplotlib is one of the most widely used data visualization libraries in Python.



Search in course

Search



Tushar Raha

Data Visualization with Python > Week 1 > Summary: Introduction to Data Visualization Tools

< Previous Next >

30 items

✓ Ungraded App Item: Hands-on Lab: Introduction to Matplotlib and Line Plots  
20 min

✓ Practice Quiz: Practice Quiz: Introduction to Data Visualization Tools  
5 questions

✓ Reading: Summary: Introduction to Data Visualization Tools  
10 min

Ungraded Plugin: Cheat Sheet: Data Preprocessing Tasks in Pandas & Plot Libraries  
10 min

Quiz: Graded Quiz: Introduction to Data Visualization Tools  
10 questions

# Summary: Introduction to Data Visualization Tools

Congratulations! You have completed this module. At this point in the course, you know:

- Data visualization is the process of presenting data in a visual format, such as charts, graphs, and maps, to help people understand and analyze data easily.
- Data visualization has diverse use cases, such as in business, science, healthcare, and finance.
- It is important to follow best practices, such as selecting appropriate visualizations for the data being presented, choosing colors and fonts that are easy to read and interpret, and minimizing clutter.
- There are various types of plots commonly used in data visualization.
- Line plots capture trends and changes over time, allowing us to see patterns and fluctuations.
- Bar plots compare categories or groups, providing a visual comparison of their values.
- Scatter plots explore relationships between variables, helping us identify correlations or trends.





Search in course

Search



Tushar Raha

Data Visualization with Python > Week 1 > Summary: Introduction to Data Visualization Tools

< Previous Next >

30 items

✓ Ungraded App Item: Hands-on Lab: Introduction to Matplotlib and Line Plots  
20 min

✓ Practice Quiz: Practice Quiz: Introduction to Data Visualization Tools  
5 questions

✓ Reading: Summary: Introduction to Data Visualization Tools  
10 min

Ungraded Plugin: Cheat Sheet: Data Preprocessing Tasks in Pandas & Plot Libraries  
10 min

Quiz: Graded Quiz: Introduction to Data Visualization Tools  
10 questions

- Box plots display the distribution of data, showcasing the median, quartiles, and outliers.
- Histograms illustrate the distribution of data within specific intervals, allowing us to understand its shape and concentration.
- Matplotlib is a plotting library that offers a wide range of plotting capabilities.
- Pandas is a plotting library that provides Integrated plotting functionalities for data analysis.
- Seaborn is a specialized library for statistical visualizations, offering attractive default aesthetics and color palettes.
- Folium is a Python library that allows you to create interactive and customizable maps.
- Plotly is an interactive and dynamic library for data visualization that supports a wide range of plot types and interactive features.
- PyWaffle enables you to visualize proportional representation using squares or rectangles.
- Matplotlib is one of the most widely used data visualization libraries in Python.
- Matplotlib was initially developed as an EEG/ECoG visualization tool.
- Matplotlib's architecture is composed of three main layers: Backend layer, Artist layer, and the Scripting layer.





Search in course

Search



Tushar Raha

Data Visualization with Python > Week 1 > Summary: Introduction to Data Visualization Tools

< Previous Next >

30 min

✓ Ungraded App Item: Hands-on Lab: Introduction to Matplotlib and Line Plots  
20 min

✓ Practice Quiz: Practice Quiz: Introduction to Data Visualization Tools  
5 questions

✓ Reading: Summary: Introduction to Data Visualization Tools  
10 min

Ungraded Plugin: Cheat Sheet: Data Preprocessing Tasks in Pandas & Plot Libraries  
10 min

Quiz: Graded Quiz: Introduction to Data Visualization Tools  
10 questions

- The anatomy of a plot refers to the different components and elements that make up a visual representation of data.
- Matplotlib is a well-established data visualization library that can be integrated in different environments.
- Jupyter Notebook is an open-source web application that allows you to create and share documents.
- Matplotlib has a number of different backends available.
- You can easily include the label and title to your plot with plt.
- In order to start creating different types of plots of the data, you will need to import the data into a Pandas DataFrame.
- A line plot is a plot in the form of a series of data points connected by straight line segments.
- Line plot is one of the most basic type of chart and is common in many fields.
- You can generate a line plot by assigning "line" to 'Kind' parameter in the plot() function.

✓ Completed

Go to next item



[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Due Oct 1, 11:59 PM IST

Graded Quiz • 30 min

## Congratulations! You passed!

Grade received 100% Latest Submission Grade 100% To pass 70% or higher

[Go to next item](#)

1. Who was the creator of Matplotlib?

1 / 1 point

- Cleve Moler, an American mathematician and computer programmer
- James Gosling, a Canadian computer scientist
- Daniel Johnson, a German physicist
- John Hunter, an American neurobiologist



Correct

Correct! The creator of Matplotlib was John Hunter, an American neurobiologist.

2. Using the inline backend, at what point can you not modify a figure?

1 / 1 point

- Before it is rendered

[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Graded Quiz • 30 min

Due Oct 1, 11:59 PM IST

2. Using the inline backend, at what point can you not modify a figure?

1 / 1 point

- Before it is rendered
- After it is rendered
- After it is created
- After it is coded



Correct

Correct! One limitation of this backend is that you cannot modify a figure once it's rendered. So, after rendering the figure, we cannot add, for example, a figure title or labels to its axes.

3. Using Matplotlib magic functions which code starts the command?

1 / 1 point

- %matplotlib notebook
- \$matplotlib outline
- %matplotlib inline

[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Due Oct 1, 11:59 PM IST

Graded Quiz • 30 min

3. Using Matplotlib magic functions which code starts the command?

1 / 1 point

- %matplotlib notebook
- \$matplotlib outline
- %matplotlib inline
- %matplotlib



Correct

Correct! The command starts with "%matplotlib," and notebook is one of the Matplotlib backends. A sign of a magic function is that it starts with "%matplotlib."

4. What does a line plot display?

1 / 1 point

- Information as a series of data points connected by straight lines.
- A line plot displays information through bars on a chart
- A plot that displays line fragments at different data points
- A line plot is used to display information on pie charts

[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Graded Quiz • 30 min

Due Oct 1, 11:59 PM IST

## 4. What does a line plot display?

1 / 1 point

- Information as a series of data points connected by straight lines.
- A line plot displays information through bars on a chart
- A plot that displays line fragments at different data points
- A line plot is used to display information on pie charts

**Correct**

Correct! Line plots display information as a series of data points connected by straight lines.

## 5. Complete the following. Matplotlib's three main layers are:

1 / 1 point

- Backend, Artist, Scripting
- FigureCanvas, Renderer, Event
- Artist, Scripting, Histogram
- UI-Loop, Drawing, Cache

[Back](#) Graded Quiz: Introduction to Data Visualization Tools  
Graded Quiz • 30 min

Due Oct 1, 11:59 PM IST



Correct! Line plots display information as a series of data points connected by straight lines.

## 5. Complete the following. Matplotlib's three main layers are:

1 / 1 point

- Backend, Artist, Scripting
- FigureCanvas, Renderer, Event
- Artist, Scripting, Histogram
- Line2D, Rectangle, Circle



Correct! Matplotlib's three main layers are Backend Layer, Artist Layer, and Scripting Layer.

## 6. What is Jupyter Notebook?

1 / 1 point

- It is a tool used for creating conventional visualization tools using the plot function
- An open-source web application that allows you to create and share documents that contain live code, visualizations, and some explanatory text as well
- A well-established data visualization library that can be integrated into different environments

[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Due Oct 1, 11:59 PM IST

Graded Quiz • 30 min

## 6. What is Jupyter Notebook?

1 / 1 point

- It is a tool used for creating conventional visualization tools using the plot function
- An open-source web application that allows you to create and share documents that contain live code, visualizations, and some explanatory text as well
- A well-established data visualization library that can be integrated into different environments
- A Python library with a number of different backends available



Correct

Correct! Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, visualizations, and some explanatory text as well.

## 7. True or False. Matplotlib was initially developed as an EEG and ECoG visualization tool.

1 / 1 point

- True
- False



Correct

Correct! Matplotlib was initially developed as an EEG and ECoG visualization tool.

[Back](#) Graded Quiz: Introduction to Data Visualization Tools

Due Oct 1, 11:59 PM IST

Graded Quiz • 30 min

8. What are the backend layers three built-in interface classes:

1 / 1 point

- Renderer, Event, and Figure
- Figure, Renderer, and Canvas
- FigureCanvas, Figure, and Event
- FigureCanvas, Renderer, and Event

 Correct

Correct! The backend layers three built-in interface classes are FigureCanvas, Renderer, and Event.

9. True or False: Line plots can be misleading if the scales on the axes are not carefully chosen to reflect the data accurately.

1 / 1 point

- True
- False

 Correct

Correct! Line plots can be misleading if the scales on the axes are not carefully chosen to reflect the data accurately. Line plots capture trends and changes over time, allowing us to see patterns and fluctuations.

[Back](#)

## Graded Quiz: Introduction to Data Visualization Tools

Graded Quiz • 30 min

Due Oct 1, 11:59 PM IST

10. Which of the following plots is not ideal for comparing different categories or groups? Select all that apply.

1 / 1 point

 Scatter plots **Correct**

Correct! scatter plots are not ideal for comparing different categories or groups.

 Pie plots **Correct**

Correct! pie plots are not ideal for comparing different categories or groups.

 Bar plots Line plots **Correct**

Correct! line plots are not ideal for comparing different categories or groups.